

# Forest Fire Surveillance System

## CS 258 - Group O

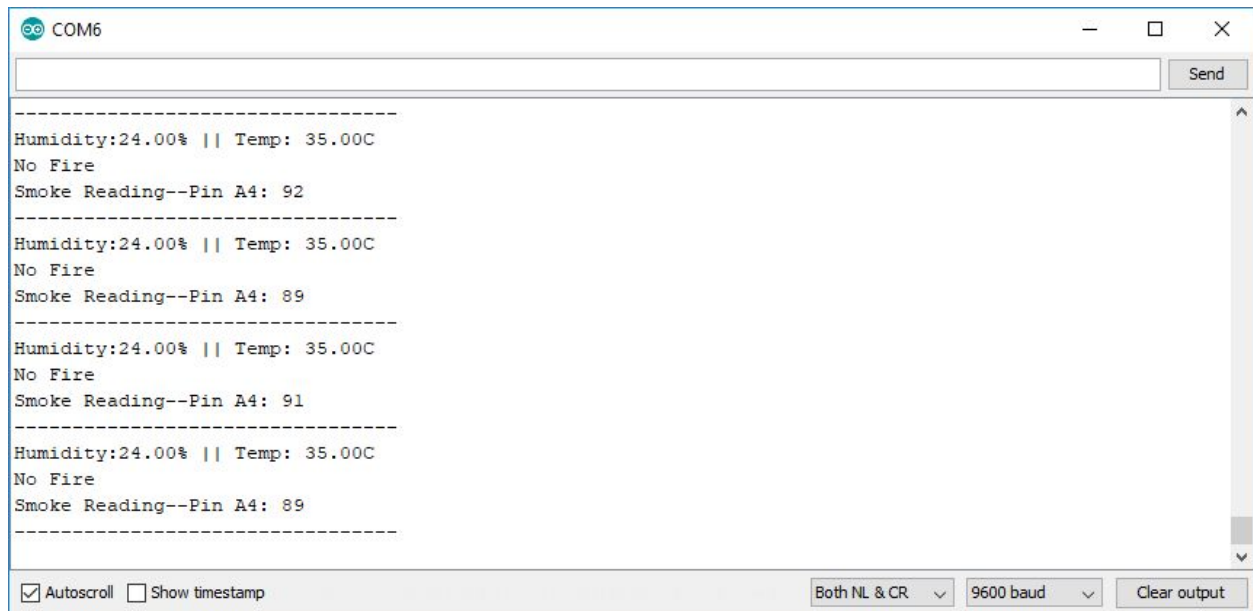
### Test Cases

Arduino-ESP8266 based IoT nodes form the backbone of the system. Various sensors viz. Flame sensor, Smoke sensor and Temperature & Humidity sensor are connected to the arduino along with the ESP8266 Wifi Module and, the data collected through the sensors is posted to the host through the ESP8266 Wifi module.

The test cases for various stages of the project progress, from collecting data through the sensors, to connecting Wifi modules with each other and then sending data between them, are as described in this document.

## 1. Collecting Data through the sensors

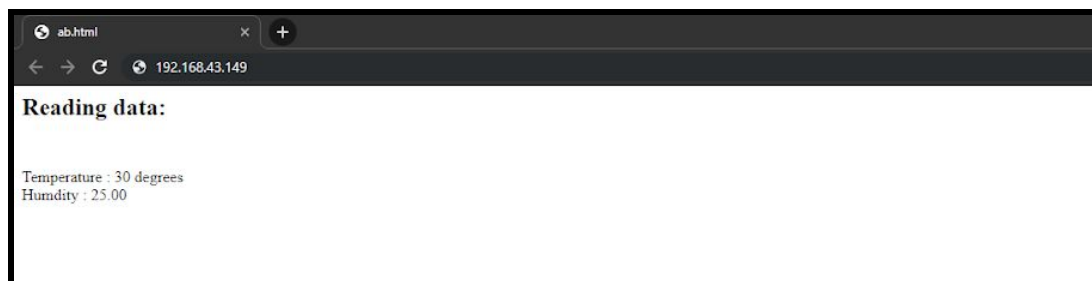
Temperature & Humidity, Smoke and Flame sensors are connected to the Arduino at different analog pins and the data collected is displayed on the Serial Monitor.



We compared the temperature data with a verified mobile utility by google . There was an absolute error of 2 °C .

## 2. Posting data on webpage through ESP8266

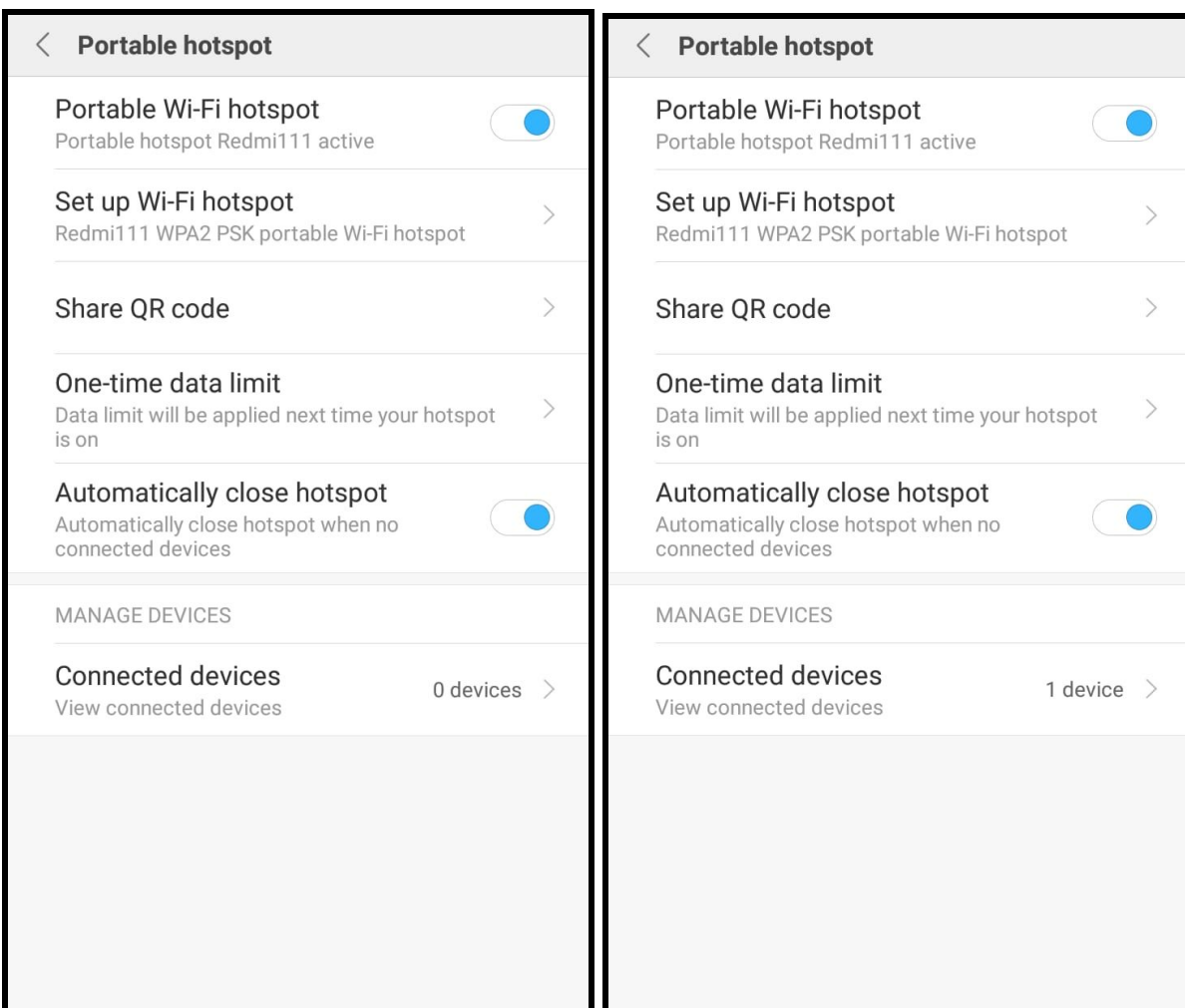
Data is collected through the sensors and then posted on the webpage at the IP address of the ESP8266 module using the preloaded AT firmware commands. The data posted can be viewed, by any device sharing the same host (which can be any mobile phone or laptop hotspot) as the ESP8266 module, on the web page at the IP address of the ESP8266 module.



We printed the data on the Serial Monitor and viewed the data on the hosted web page. Both gave the same output . Hence we can say the data was successfully posted to the web page.

### 3. Successfully Connecting to the WiFi

We uploaded code onto the WiFi module that was meant to establish connection between a hotspot and WiFi module. In this case, we established the hotspot via a mobile device. In the images below we can see that as soon as the WiFi module connects to the WiFi hotspot , the number of connected devices on the phone changes from 0 to 1.



## 4. Setting up web server on ESP8266

Now, custom code is uploaded on the ESP8266 to host web pages, as the pre-uploaded AT firmware has limited functionality. The code allows the ESP8266 to host a web page on its IP address. Any device connected to that the same network as the WiFi module should be able to view the data posted at the web page. We connected to the host on a laptop device and we could see the output on the web page.



## 5. Sending data from one ESP8266 module to another

Now, the two ESP8266 modules are made to connect to each other by making one as the host and other its client. The host should be able to fetch data from the get request. We sent some particular data via a GET request by connecting a laptop to the WiFi module's hotspot and it was printed successfully on the Serial Monitor of module.

Hence we can say the data was successfully transferred to the host module. We opened the following URL -

<http://192.168.4.1/data?value=Sample%20Data%Print>

The output received matched with the string sent

COM9

Send

```
load 0x4010f000, len 1384, room 16
tail 8
chksum 0x2d
csum 0x2d
v951aeffa
~ld

Configuring access point...AP IP address: 192.168.4.1
HTTP server started
Sample Data Print
Arjit
data going via get
```

☐ Autoscroll ☐ Show timestamp

Both NL & CR 115200 baud Clear output